Prediction Assignment WriteUp

Marian SVatko

11/9/2017

Summary

The goal of the project was to predict the type of exercise (outcome) based on 53 variables (predictors).

Data Pre-Processing

The first 7 variables (such as username, timestamp, etc.) did not appear to be related to the outcome variable and they were deleted. In, addition, considerable NA values were present in the dataset. These values, together with #DIV/0! values were assigned zero values. Incomplete rows were also deleted. in addition, predictors that have one unique value (i.e. are zero variance predictors) were also removed from dataset before further processing.

```r
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
## Warning in as.POSIXlt.POSIXct(Sys.time()): unknown timezone 'zone/tz/2017c.
## 1.0/zoneinfo/Europe/Vienna'
```

```r
options(warn = -1)
setwd("/Users/mariansvatko/Downloads")
training <- read.csv("pml-training.csv")
testing <- read.csv("pml-testing.csv")
training <- training[ ,-c(1:7)]
testing <- testing[ ,-c(1:7)]
training[training=="#DIV/0!"] <- 0
testing[testing=="#DIV/0!"] <- 0
training[is.na(training)] <- 0
testing[is.na(testing)] <- 0
training <- training[complete.cases(training),]
training <- training[,-nearZeroVar(training)]
```

Prediction

Considering 5 different values of the predicted variable and the calculation speed, K-nearest neighbor prediction method was selected.

```r
set.seed(447)
inTrain <- createDataPartition(y=training$classe, p=0.7, list=FALSE)
train <- training[inTrain,]
test <- training[-inTrain,]
train$classe <- as.factor(train$classe)
test$classe <- as.factor(test$classe)
preP <- preProcess(train[,-length(train)], method=c("scale","center","pca"), thresh=0.9)
fit <- train(classe ~., data = predict(preP, train), method = "knn")
```

```
prediction <- predict(fit, predict(preP, test))
confusionMatrix(prediction, test$classe)$overall[1]
```

```
##  Accuracy
## 0.9531494
```

```
confusionMatrix(prediction, test$classe)$table
```

```
##           Reference
## Prediction    A    B    C    D    E
##          A 1606   40    3    2    6
##          B    7 1030   25    3   10
##          C   17   32  955   61    8
##          D   10    9   18  875    7
##          E    1    4    4    3 1027
```

```
testing$classe <- predict(fit, predict(preP, testing))
```

Conclusion

After bulding the prediction model, it was evaluated on the testing data. 95.31% accuracy was achieved. Machine learning algorithm was applied to the 20 test cases. The results are as follows:

```
testing$classe
```

```
##  [1] B A A A A E D B A A D C B A E E A B B B
## Levels: A B C D E
```